

INFORME FINAL



Comedero Automático con Reconocimiento

Electrónica y Automatización

Narella Rojas

Fecha de inicio: **17/10/2025**

Fecha de entrega del informe: **20/10/2025**

Contenido

1.	Introducción	2
2.	Objetivos SMART.....	4
3.	Fase de Diseño	5
a.	Requerimiento de Hardware y Software	5
b.	Diagrama de Conexiones.....	7
c.	Caso de Uso	8
d.	Diagrama de Máquina de Estados.....	9
e.	Listado de Tareas a Realizar	10
4.	Presupuesto	11

1. Introducción

En hogares con múltiples mascotas que requieren dietas especializadas, surge un problema crítico de **control de acceso a la alimentación**. Específicamente, en mi caso, tres gatos conviven con necesidades dietéticas estrictas que, al cruzarse, generan riesgos para su salud.

- **Julieta (16 años):** Requiere exclusivamente comida **renal** debido a una condición crónica.
- **Robin (9 años):** Requiere comida **hepática** y renal. La ingesta de comida normal o un exceso de la dieta de Julieta podría causarle problemas.
- **Chimuelo (7 años):** Gato sano que, por ética veterinaria y nutricional, no debe consumir las dietas terapéuticas de alto costo y composición especializada de los otros gatos.

La necesidad principal es **garantizar que cada gato solo consuma la comida asignada**, eliminando la supervisión constante y el riesgo de que las dietas se mezclen o sean ingeridas por la mascota equivocada.

El proyecto busca resolver el problema de la alimentación cruzada mediante la automatización y la identificación individual. El objetivo es diseñar e implementar un **Comedero Inteligente por Reconocimiento RFID** que actúe como un **portero automático de la comida**.

Se desarrollará un sistema de control de acceso que:

1. **Identifique al animal** que se acerca al plato utilizando tecnología **RFID (Identificación por Radiofrecuencia)**, leyendo el *llavero tag* en el collar asociado a cada gato.
2. **Verifique los permisos** de ese animal para la dieta específica contenida en el comedero.
3. **Accione un mecanismo de apertura/cierre** (controlado por un servomotor) para desbloquear el acceso al plato **solo si** el animal tiene el permiso requerido.

Esto resultará en la **eliminación del riesgo de intoxicación o empeoramiento de condiciones médicas crónicas** debido a la ingesta errónea de alimentos, a la vez que se **reduce el estrés y la carga de trabajo** del cuidador.

La implementación de este proyecto involucra directamente a varias partes interesadas:

- **Clientes Internos (El Cuidador/Yo):** El principal beneficiario. El sistema le proporciona **paz mental, automatización** de una tarea diaria intensiva (vigilancia), y un **registro potencial** del consumo de alimentos (a futuro), optimizando el tiempo y reduciendo el desperdicio de comida especializada.
- **Clientes Externos (Los Gatos: Julieta, Robin y Chimuelo):** Son los usuarios finales del sistema. Su participación es pasiva, pero crítica, ya que el proyecto garantiza su **salud y bienestar** al asegurar el cumplimiento estricto de sus prescripciones dietéticas.

- **Expertos/Asesores (Veterinarios):** Aunque no participan en el desarrollo del proyecto directamente, se podrían beneficiar indirectamente de la **fiabilidad del sistema** para asegurar que los tratamientos dietéticos complejos se sigan al pie de la letra, mejorando los pronósticos de los pacientes.

2. Objetivos SMART

- I. **Específico:** Implementar un prototipo de **Comedero Inteligente Automático** mediante **Arduino UNO** que utiliza el sistema **RFID** para **identificar a cada mascota** y controlar de forma independiente el acceso a su plato de dieta especializada a través de servomotores.
- II. **Medible:** El sistema deberá: **1.** Lograr una en la verificación de permisos altas, **2.** Abrir el acceso en **menos de 3 segundos** tras la lectura correcta del ID, y **3.** Cerrar el acceso en **menos de 5 segundos** tras la ausencia del gato.
- III. **Alcanzable:** Sí, el proyecto es alcanzable utilizando la electrónica de bajo costo y las librerías estándar de Arduino UNO, lo que lo hace totalmente factible en el laboratorio.
- IV. **Relevante:** El proyecto es altamente relevante al resolver un problema crítico de salud animal (alimentación cruzada de dietas renales y hepáticas), automatizando completamente la supervisión del cuidador.
- V. **Plazo:** El desarrollo total del sistema, incluyendo el diseño físico, la electrónica y la programación de la lógica de los tres platos, será completado y demostrado antes de la fecha límite del **7 de noviembre**.

3. Fase de Diseño

a. Requerimiento de Hardware y Software

El desarrollo del Comedero Inteligente por Reconocimiento RFID se basa en una arquitectura de microcontrolador, aprovechando la simplicidad y robustez del **Arduino UNO** como núcleo del sistema.

En el ámbito del **Hardware**, se requieren tres componentes principales que aseguran la funcionalidad del sistema: la identificación, la detección y la actuación.

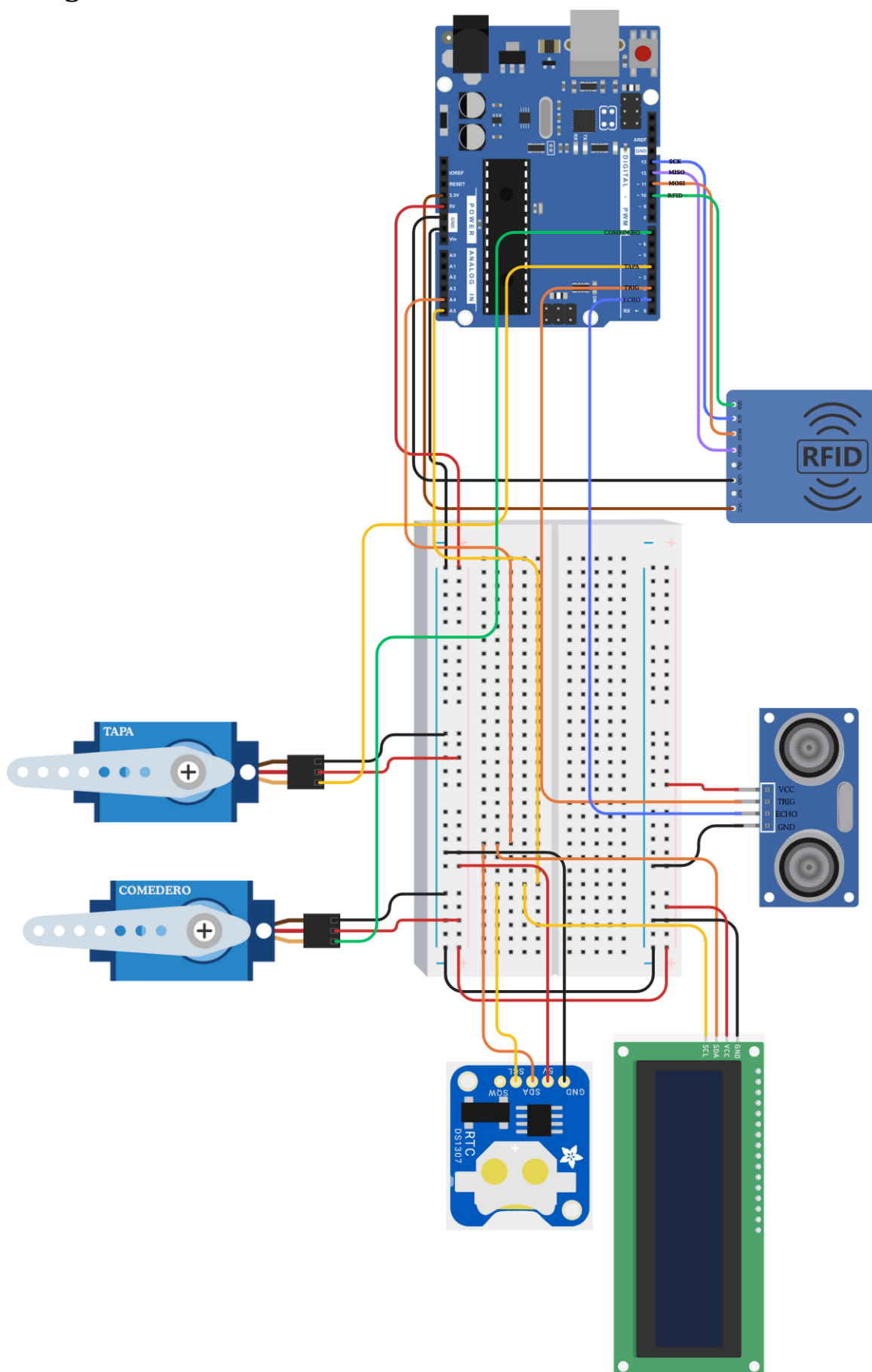
1. **Identificación:** Se utilizará el **Módulo Lector RFID RC522** junto con sus respectivas **Etiquetas (Tags)**. El lector es crucial para leer el ID único de cada gato al acercarse al plato, y estos IDs actúan como la clave de acceso para autorizar o denegar la alimentación.
2. **Detección:** La presencia del gato frente al comedero será detectada por un **Sensor Ultrasónico HC-SR04** (o un sensor IR de proximidad). Este sensor es fundamental para activar el sistema y minimizar el consumo de energía, iniciando el proceso de lectura RFID solo cuando es necesario.
3. **Actuación:** El control físico de la tapa de acceso a la comida se logrará mediante un **Servomotor de Alto Torque** (como el MG996R). La elección de un servo de alto torque es necesaria para garantizar el movimiento rápido y fiable de la tapa, incluso ante posibles fricciones.
4. **Automatización de servicio:** Se incorporará un **Módulo de Reloj en Tiempo Real (RTC)**, como el **DS3231**. Este componente mantiene la hora y fecha exactas independientemente del Arduino (usando una batería pequeña), lo cual es esencial para programar las **tres raciones de comida** diarias a horas fijas y automatizar el inicio del ciclo de dispensación.
5. **Interfaz:** Se integrará una **Pantalla LCD**, este display es crucial para informar al cuidador y al propio sistema sobre el estado actual:
 - Mostrar la hora y fecha del RTC.
 - Indicar mensajes de estado ("Esperando mascota...", "Acceso permitido", "Acceso denegado", "Comedero listo. Esperando mascota...").
 - Informar el estado de la dosificación ("Hora de comer: Llenando plato...").

Para la **alimentación**, es vital contar con una **Fuente de Alimentación Externa de 5 V y al menos 2 A**. Aunque el Arduino UNO puede alimentarse por USB, el servomotor requiere un suministro de corriente estable y alto que no debe depender del microcontrolador, previniendo reinicios o fallas operacionales.

En cuanto al **Software**, el proyecto se desarrollará en el **Arduino IDE** utilizando dos librerías esenciales. La librería **MFRC522.h** es indispensable para establecer la comunicación mediante el protocolo SPI con el lector RFID y gestionar la lectura de los IDs. Paralelamente, la librería **Servo.h** simplificará la programación de los movimientos de apertura y cierre del servomotor. Para la funcionalidad de automatización, se requerirá la librería **RTClib.h** para leer

la hora y fecha del módulo de reloj y gestionar la programación de las tres dosis diarias de alimento. Finalmente, la librería para el control de la pantalla **LiquidCrystal_I2C.h** será crucial para comunicar los mensajes de estado en tiempo real ("Acceso Permitido", "Hora de Comer") al usuario. La lógica central del programa unirá estas librerías para implementar el diagrama de máquina de estados definido, asegurando que solo los IDs permitidos activen el actuador y que la dispensación ocurra únicamente a las horas programadas.

b. Diagrama de Conexiones



c. Caso de Uso

Si tomamos el **plato de Julieta** (únicamente comida **renal**) como ejemplo, los casos serían así:

Caso de Acceso Otorgado: La secuencia se inicia con la **Detección de Proximidad**; el gato (Julieta, en este caso) se acerca al comedero, lo que activa el sensor ultrasónico (o IR), en la pantalla aparecerá "Mascota detectada". Al detectar la presencia del animal a una distancia inferior a los 20 cm, el sistema sale de su estado de espera.

A continuación, el **Sistema** activa el módulo lector **RFID RC522** y comienza a escanear. Julieta, al acercar su cabeza para acceder a la comida, aproxima el tag RFID de su collar al lector. Una vez que el lector capta y transmite el **ID único de Julieta** (ID_JULIETA) al Arduino, se procede a la **Verificación de Permisos**.

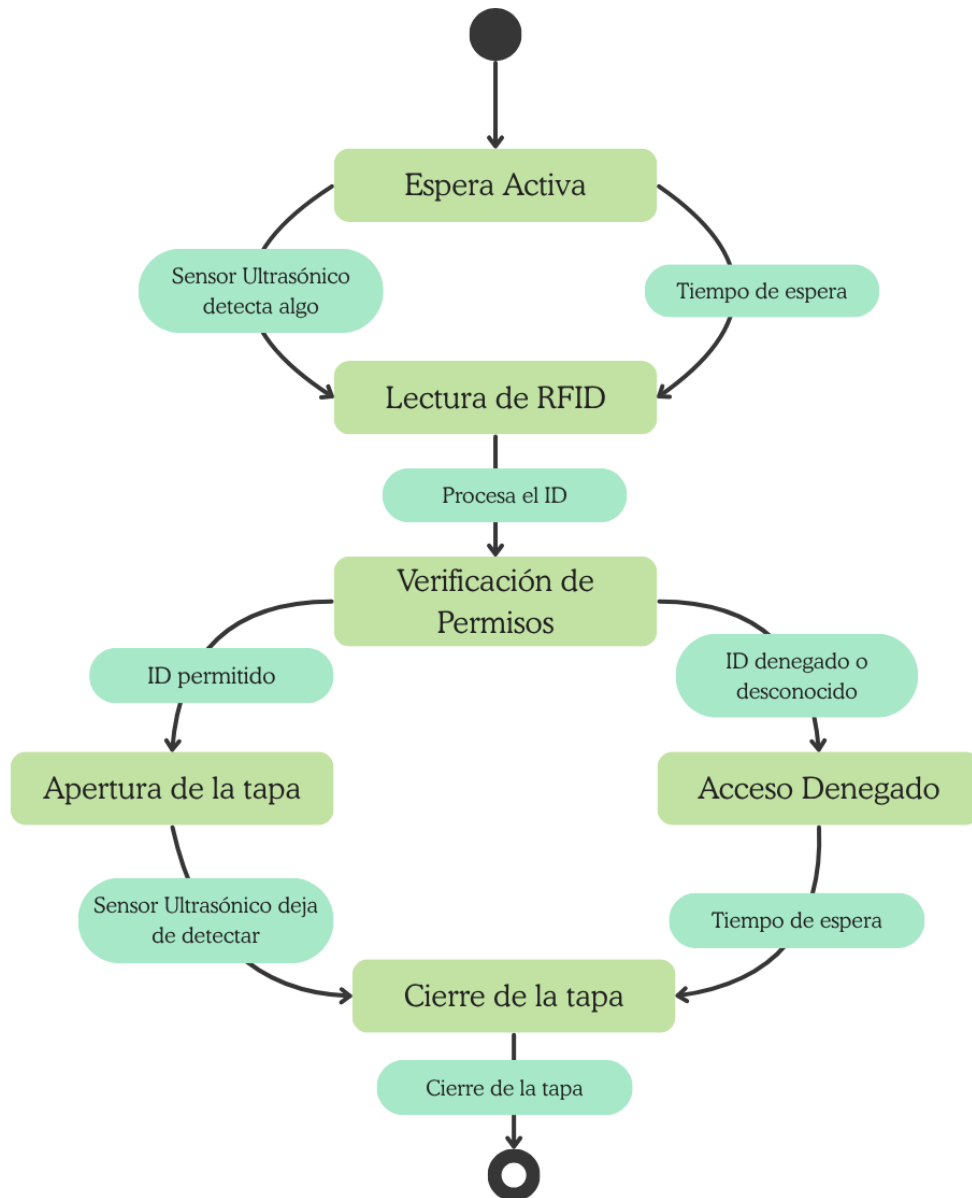
La **Lógica de Código** compara el ID leído con la lista de IDs autorizados para este plato específico (en este caso, solo el ID_JULIETA). Como el ID coincide con uno de los permisos concedidos, el sistema activa el **Servomotor**, el cual gira a la posición de **ABIERTO** (90°), mostrando en la pantalla "Acceso permitido", retirando la tapa y permitiendo a Julieta el consumo de su dieta renal.

Julieta consume la comida, mientras el sensor ultrasónico mantiene el plato abierto al seguir detectando su presencia. Cuando el gato se aleja, el sensor deja de detectar proximidad. En ese momento, el sistema inicia una cuenta regresiva de seguridad (5 segundos) y, al finalizar, ordena al Servomotor que gire a la posición de **CERRADO** (0°), en la pantalla aparecerá "Comedero listo. Esperando mascota..." cubriendo el plato y previniendo el acceso de otros gatos.

Caso de Acceso Denegado: Si **Chimuelo** (o Robin, cuyos IDs no están en la lista de permisos) se acerca, la lógica de código identifica que el ID_CHIMUELO no tiene acceso al plato renal. Por lo tanto, el Servomotor **permanece en la posición de CERRADO** y mostrará el mensaje de "Acceso denegado", garantizando la seguridad de la dieta y la integridad de la comida especializada.

el sistema incorpora un Módulo de Reloj en Tiempo Real (RTC), como el DS3231, para automatizar el llenado del plato tres veces al día. La Lógica de Código verificará la hora actual con el RTC y, al coincidir con uno de los horarios programados, el sistema se activará para realizar la **administración automática** de la porción de alimento en el plato, mostrando en la pantalla el mensaje "**Hora de comer: Llenando plato...**". Si la dosificación ya se realizó, la pantalla indicará "**Comedero listo. Esperando mascota...**".

d. Diagrama de Máquina de Estados



e. Listado de Tareas a Realizar

1. **Adquisición de componentes:** Comprar y reunir todos los componentes de hardware necesarios (Arduino UNO, 3 módulos RC522, 3 servomotores, 3 sensores HC-SR04, tags RFID y la fuente de alimentación externa).
2. **Prueba del RC522:** Poner a prueba la funcionalidad de los lectores RFID. Leer y registrar el ID único de los 3 tags RFID (uno por gato: Julieta, Robin, Chimuelo) para usarlos en el código.
3. **Prueba del servomotor:** Poner a prueba los servomotores. Implementar la librería *Servo.h* y programar las posiciones extremas de **Apertura** (90°) y **Cierre** (0°) para el mecanismo de la tapa.
4. **Prueba del HC-SR04:** Poner a prueba los sensores de proximidad. Programar la función de detección de proximidad (estableciendo la distancia de activación, por ejemplo, < 20 cm).
5. **Prueba del reloj:** Integrar y probar el Módulo RTC. Configurar la hora y la fecha iniciales y programar la función de verificación horaria para los tres momentos de dispensación de la comida (8 AM, 2 PM, 8 PM).
6. **Prueba de pantalla:** Integrar y probar la Pantalla LCD. Incluir las librerías *LiquidCrystal_I2C.h* programar las funciones para mostrar mensajes de estado (Hora, "Mascota detectada", "Acceso Denegado").
7. **Diseño de la estructura:** Diseñar el prototipo físico. (Viendo)
8. **Armar el circuito:** Realizar el montaje electrónico final. Conectar todos los componentes al Arduino UNO.
9. **Escribir el código:** Unir todas las funciones y desarrollar la lógica completa de la Máquina de Estados.
10. **Pruebas y calibración:** Probar el sistema integrado. Calibrar el tiempo de lectura RFID, la posición exacta de los servomotores, la sensibilidad de los sensores de proximidad y que los mensajes se muestren correctamente en la pantalla. También el sistema integrado en un **ciclo completo de 24 horas** (para verificar el RTC). Calibrar tiempos y posiciones.

4. Presupuesto

Componente	Precio por Unidad	Cantidad	Precio Final
Arduino + Cable	\$15.000	1	\$15.000
Módulo Lector RFID RC522 + Tags	\$6.000	1	\$6.000
Servomotor	\$5.000	2	\$10.000
Sensor Ultrasónico HC- SR04	\$3.500	1	\$3.500
Modulo reloj	\$3.200	1	\$3.200
LCD 16x2	\$6.990	1	\$6.990
Cartón o plástico	\$0	x	\$0
TOTAL			\$44.690

Fuente: Mercado Libre